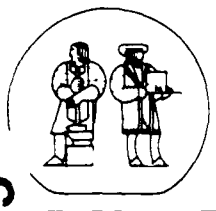


FILE COPY

APPROVED FOR PUBLIC RELEASE
DISTRIBUTION UNLIMITED

AD-A213 005

COMPUTER-AIDED FABRICATION SYSTEM IMPLEMENTATION

Final Technical Report for the period April 1, 1985 to December 31, 1988

Massachusetts Institute of Technology
Cambridge, Massachusetts 02139

Principal Investigators:	Paul Penfield, Jr.	(617) 253-2506
	Dimitri A. Antoniadis	(617) 253-4693
	Stanley B. Gershwin	(617) 253-2149
	Emanuel M. Sachs	(617) 253-5831
	Stephen D. Senturia	(617) 253-6869
	Donald E. Troxel	(617) 253-2570

DTIC
ELECTE
OCT 02 1989
S D

This research was sponsored by Defense Advanced Research Projects Agency (DoD), monitored by Clifford Lau through the Office of Naval Research under ARPA Order No. 5339, Contract No. N00014-85-K-0213.

89 10 2 04 7

I. Introduction

The Massachusetts Institute of Technology is pleased to submit this final report for contract N00014-85-K-0213, awarded by the Information Science Technology Office of the Defense Advanced Research Projects Agency, as monitored by the Office of Naval Research. The contract, and this report, cover the period between April 1, 1985 and December 31, 1988.

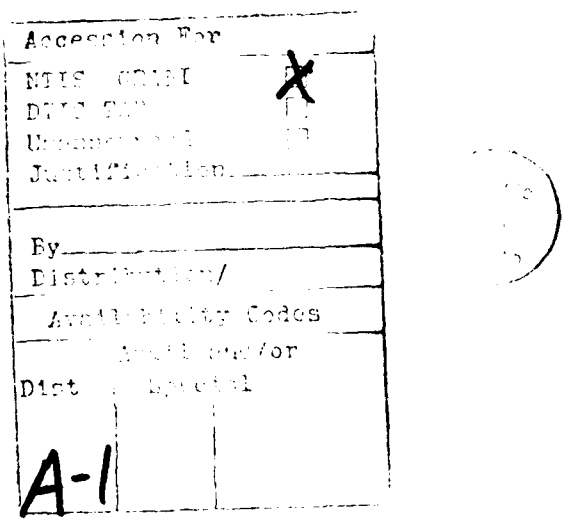
The major objective of this contract was to develop a computer system to aid the fabrication of VLSI integrated circuits. Particular attention was paid to use of university and industry standards in the system, and an open architecture. Many of the ideas behind such a system were articulated by an inter-university group in an unpublished memo which subsequently was issued as an MIT VLSI memo: P. Penfield, Jr., S. B. Gershwin, D. A. Hodges, C. M. Osburn, J. Reynolds, J. Schott, A. J. Steckl, and D. E. Troxel, "Requirements for Computer-Aided Fabrication," MIT VLSI Memo No. 84-200, January 17, 1984.

Other objectives of this contract were to extend the system architecture to encompass mechanical properties of integrated circuits, particularly as these may affect reliability, and to incorporate models for VLSI fabrication equipment.

The principal investigator was Professor Paul Penfield, Jr. Work was performed in the following five areas, under the direction of the individuals indicated:

- CAF System Structure (Professor Donald E. Troxel)
- Modular Process (Professor Dimitri A. Antoniadis)
- Scheduling (Dr. Stanley B. Gershwin)
- Equipment Modeling (Professor Emanuel M. Sachs)
- Mechanical Property TCAD (Professor Stephen D. Senturia)

The remainder of this report discusses progress made in each of these areas. For further information, readers are referred to the publications listed in the final section.



II. CAF System Structure

The objective of this task, the largest one in this contract, was to design, develop, and implement a CAF system and deploy it in various MIT fabrication facilities. The system was named CAFE (Computer Aided Fabrication Environment). It was designed to support more than one facility, and indeed data from three separate facilities was put into the system. The facility that relies most heavily on this work in its daily operation is the Integrated Circuit Laboratory (ICL).

In this activity, the most difficult challenges were (1) the design and implementation of a data model that was independent of particular data base managers and presented a constant interface to application programs; and (2) the design of a process flow representation, that is, a language in which to express the fabrication sequence. These two tasks formed the basis of two doctoral theses. A challenge of a different kind was to implement and maintain a hardware and software system for routine daily use. Without such a system, of course, none of the research under this contract could proceed, yet this activity required more development than research, and staff members were employed to carry it out. Several bachelors and masters theses were based on portions of this work.

The progress of this activity is chronicled below by quoting from the semiannual reports prepared for each DARPA VLSI Contractor's Meeting, held each spring and fall.

Fall 1985

We have implemented a preliminary version of a CAF system for use in our fabrication facility. The architecture and capabilities of this system, named CAFE, Computer-Aided Fabrication Environment, were defined during summer 1985, and the system was ready for brave users (who don't mind a few bugs) on September 1, 1985.

The desired ultimate set of capabilities was described over a year ago by a multi-university group. During the summer a selection was made of features to be implemented first. The features with the highest perceived immediate value were not those of ultimate scientific and engineering interest, so much as those that enabled the new facility to start off immediately without paper. This was deemed important in setting people's attitudes toward contamination right from the beginning. The specific features selected for the initial implementation were:

User List
Machine list
User Qualification Table
Message-of-the-day
Personal Laboratory Notebook
Machine Operating Instructions
Activity Log

The hardware features deemed most urgent (aside from a central computer to run the system) were terminals in key technicians' and engineers' offices, a terminal in the gowning room, and terminals in the clean room.

The approach taken in this first implementation is to do as much as possible by using UNIX shell scripts, and other standard UNIX features. The user list was implemented by expanding the Unix password file. Each machine is a directory in the data area, and therefore the machine list is merely a directory listing. The message-of-the-day is the standard UNIX facility. The laboratory notebooks were implemented by a relatively simple shell script which merely appends terminal input to a file. The data structures are all file-based. No attempt was made to use a relational data base at this early stage.

At this time CAFE is being taught to the first group of users. It appears adequate to support the paperless feature of the lab, but is still very crude.

Spring 1986

Prior to the beginning of this reporting period, an initial, rudimentary CAF system was designed and implemented. This included a personal notebook and data structures for the upcoming fabrication equipment installation. This was implemented on a borrowed VAX 11/750 and some of the engineering staff started to use the lab notebook and editor. RS232 cables to the offices and labs were installed and some were checked and connected to terminal concentrator ports. Three concentrators were assembled, connected to an MIT Ethernet, and were operational. The VAX 11/785 hardware was installed.

Since then, we have completed the computer hardware installation, evaluated and purchased terminals for both users and for use in the clean rooms, installed Unix version 4.3BSD, integrated local networking software, adapted mail software to our local needs, ported several applications programs to our CAF computer, instituted a weekly back-up procedure, substantially improved our terminal concentrators, created user documentation, interfaced to the MIT physical plant computer which monitors a large number of sensing points in our facility, initiated some student projects, initiated plans for including the teaching laboratories in our developing CAF system, acquired both documentation and software for the present Berkeley CAF system, initiated acquisition of additional main memory and disk storage, and expanded our user base to include virtually all of the faculty, students, and technical staff associated with our fabrication facility.

Our present computer hardware consists of a VAX 11/785 with 8 Mbytes of main memory, two 450 Mbyte disks, GCR tape, 1600-cpi tape, laser printer, four phone lines and modems, and an Ethernet port to the MIT network. Additionally there are seven 11/73-based terminal concentrators, each of which has 32 RS232 ports. RS232 cables have been installed throughout our building, and numerous user terminals and PCs have been connected.

We have completed an extensive evaluation of terminals for use in the clean rooms, and by the technical staff and other CAF users. The information gleaned during our selection process has been communicated to other building occupants for their consideration. We chose Ergo 301 terminals as they had the best compromise considering cost, video quality, compatibility with existing software, and the fact that they do not have fans, which allows their use in the clean room areas.

We installed the latest version of UNIX, Berkeley 4.3BSD, and have integrated local networking software and have adapted the mail software to enable local as well as remote communications. Our official host name is CAF.MIT.EDU; locally the computer is known simply as CAF.

Several large applications programs have been successfully ported to run on CAF, including MAGIC and PISCES. The success in porting these programs has resulted in greater use of the caf system so much that we will soon need increased primary memory and user disk space.

A back-up regimen has been initiated with full dumps taken every month and incremental dumps every week. This is intended primarily as protection against catastrophic disk failure.

Experiments with our terminal concentrators revealed service deficiencies when all ports were active. The cause was traced to inadequate use of the available buffer memory by the SWITCH software. This has now been corrected and satisfactory performance is achieved with 32 active lines per concentrator. We are reasonably confident that we could implement 48 lines per concentrator but are not sure that satisfactory response times would be preserved if all 48 lines were active at once. We have more physical memory than can now be used, probably enough to service twice the number of lines per concentrator. However, it is doubtful that the rather substantial software rewrite required to use this memory is worth the effort.

We have automated the procedures for applying for and approving new computer accounts. An program named "open" has been written which enables prospective users to make requests for accounts and provide the required information. Mail is sent to an "approver" after an account has been requested and new accounts, including appropriate initialization files and directories, are automatically created after approval.

We have also been active in the generation of documentation for CAF users. User documentation now covers the use of terminal concentrators, obtaining accounts, obtaining documentation, using UNIX and

EMACS, using Kermit with IBM PCs, and using the nroff text formatter with the -me macro package. Manual entries for the Cshell with command completion and Mail utility have also been made available for distribution. We have initiated a project to define laser printer hardware and software to provide both typesetting and hard copy of arbitrary graphics.

We have interfaced to the MIT physical plant computer which monitors a large number of sensing points such as resistance of DI water, etc., in our facility. We capture all alarms printed and have provided a mechanism for remote query of the status of the monitored points. Our facilities personnel can now interrogate the monitoring computer from either office or home and efficiently direct corrective action.

We have initiated a plan to expand the availability of the CAF system to the integrated-circuit teaching laboratory. This will provide us with another active test bed for the CAF system, and will also allow indoctrination of entering graduate students (our present practice is to require new students to complete an intensive, short version of the undergraduate teaching laboratory subject).

We have advertised to our students several projects concerned with integrating measuring instruments into the CAF system. One student has started work on a project relating to a CV measuring instrument. Another project has been defined concerning software for a scanning thermometer intended for use in monitoring photolithography process parameters.

We have acquired both documentation and software for the present Berkeley CAF system. We are in the process of installing a test version of this software and will study the integration of this software into our system or the integration of our system into the Berkeley software.

We now have a substantial number of CAF users. The system response time remains quite good, but we have had to make plans to augment primary memory and available disk space. Our primary future activity is to further the development of the CAF applications software and its use in the installation and operation of equipment to be installed.

Fall 1986

The memory on the computer CAF.MIT.EDU has been upgraded from 8 Mbytes to 16 Mbytes. This was required as a number of users were starting to run large simulation and analysis programs resulting in excessive paging activity.

A floating point accelerator was installed. Originally we had not intended that this computer be used for intensive computations but, instead, our plan was to farm out these computations to other processing engines. In deference to our users we installed a floating point accelerator to maximize the performance of the computer.

We have substantially expanded our available disk space both to serve the needs of existing users and to provide room for the enhancement of our caf software system by the use of a relational data base.

A second laser printer has been acquired. This laser printer has a Postscript interface and software has been installed that allows Unix troff and plot output to be printed on this laser printer.

An additional terminal concentrator was installed in another building to service student offices.

All of our terminal concentrators have been upgraded to handle 48 lines each. We now have enough ports for all existing cables and also for the ones required for our new planned enlarged terminal room.

The Dektak profilometer has been connected together with software for transmitting screen plots to the caf computer. Software has also been developed for printing these screen plots on either a laser printer or a terminal (graphics or ASCII).

The Nanospec has been interfaced to the caf computer and initial software developed to receive the data output and control the Nanospec. More work needs to be done in order to discard extraneous data.

The 11/23 computer included in the furnace controller has been interfaced to the caf computer. Software has been provided to allow recipe development from terminals logged into the caf computer.

Prior to the existence of any caf effort, a large number of monitoring points such as fans, alarms, DI water resistance, etc. were implemented by connection to a computer operated by MIT Physical Plant. This computer was then accessed by a dedicated terminal and it also had a hard copy printer on which alarms were printed. Both the dedicated terminal lines and the printer line have been connected to the caf computer. Software has been developed which logs all alarms and also keeps a list of currently active alarms. As this data can accumulate without bound, we have initiated an automatic purging of the historical log entries that are more than 30 days old. These capabilities are actively used by lab personnel.

We have decided to use INGRES as a relational data base. We have initiated an active project to define a schema and implement it using "university" INGRES. We have investigated acquisition of a commercial version of INGRES but have held off because MIT is in the process of negotiating a campus wide license and this should reduce the acquisition cost.

We have begun to look at desirable languages in which to denote process flow specifications. Since such descriptions need to be able to be both executed and analyzed, it is inappropriate to consider flows as either 100% procedures or 100% data. We have begun to think of a LISP-based notation. We believe that such a language is now critical since it lies at the heart of almost all the functions that the CAF system is to perform. We believe that the language should firmly support, if not be designed explicitly around, the two-stage generic process-step model we have reported on earlier.

Spring 1987

Because of our success in getting faculty, staff, and students to use our computing facilities, and because of the increased use of our CAF system in actually operating the IC Laboratory, our VAX 785 (CAF.MIT.EDU) is heavily overloaded. We must augment our facilities. We evaluated several possibilities and have initiated the purchase of two Sun Microsystems 3/280 computers. One of these will be used for the actual running of the CAF system CAFE for the IC Laboratory and the other will be reserved for development of CAF system software. We also have ordered RTI INGRES for use on the Sun computers. Delivery of this equipment is expected during the first half of 1987.

We have made substantial progress in the development of a data model and schema. Our data architecture being developed is similar to the Multibase system developed at CCA. This provides a uniform query interface to data residing in multiple autonomous, heterogeneous data bases. Our current data base is distributed across two relational systems, university INGRES and PRELUDE and a hierarchical file system. PRELUDE is a fast, lightweight UNIX-style data manager. Our system is very modular and, thus far, it has been easy to incorporate new DBMSs into the system as well as move data from one data manager to another.

Our data model is the functional model with support of extended data types including various temporal types as well as inexact, interval, and null values. The schema captures several important aspects of plant and process management: fabrication facilities and equipment, users, equipment reservations, lots, lot tracking, wafers, process flow descriptions, WIP tracking, and lab activity information.

We have developed a generalized forms based user interface program, called FABFORM. This single program, when called with a parameter file, produces a terminal display and allows a user to move from field to field and enter data. The type and content of the screen display is specified by an ASCII file which is referenced by data included in the parameter file. The form may have arbitrary length and the user can scroll up or down. At present, user interface commands are much like EMACS commands. When the user exits, or saves the data, an updated parameter file is written.

Using FABFORM, we have implemented the electronic equivalent of a signup sheet for reserving fabrication equipment in our lab. This program is in daily use and stores the signup data in our data base. Several improvements for the next version of this reservation program have been initiated.

We are developing a process flow language. This language has a lisp-like form, although this need not be apparent to process design engineers. We continue to embrace the concept of the two-stage generic process model with corresponding expression of goals, wafer environment, and machine settings. Adequate machine models, yet to be developed, will provide a transformation from machine settings to wafer environments. Process models similarly provide the transformation from the wafer environment to the wafer-state goals.

The meaning of our process flow language is to be provided by several interpreters: fabrication, simulation, production scheduling, and "walk-through." We have completed an initial version of this "walk-through" interpreter. It interprets the process flow language and enables a process developer or potential user to see what will happen when the process is executed. It will, we think, prove useful in the design of a process, the communication of the process to others, and its approval by laboratory management. Substantial work has been accomplished towards a fabrication interpreter.

We have defined and begun work on a browser which allows a user to review what actually happened in the fabrication of a lot of wafers. We have yet to resolve how to browse through the future, especially when the future path is uncertain due to possible branches.

Fall 1987

During the past year we have defined and installed both a hardware configuration and an initial CAF software system architecture. Central to the hardware architecture is the use of computer networking, both to provide integration of functions performed on different computers and to provide access to these computers from the integrated circuit processing facility and also from the users' offices.

The primary computer used for the actual operation of the CAF system within the integrated circuit processing facility consists solely of a Sun 3/280 processor with disk and network interface. Our processing facilities and offices are equipped with ASCII terminals which are interfaced to diskless terminal concentrators, which in turn provide network connections to any available computer. There are eight terminal concentrators, each of which has provision for up to forty-eight 9600-baud RS232 ports. These concentrators also provide for communication paths and connection of some of the actual integrated circuit processing equipment.

Another Sun 3/280 is used as a development test bed for enhancements, tests, and debugging fixes. This insulates the users of the primary CAF system from the introduction of untested software, and it also helps to provide faster response time for the present users in that they do not have to share their CPU resources with the developers of new software. The second Sun 3/280 allows the developers of new software in particular to experiment with data base application programs without corrupting the actual data base used for ongoing processing of integrated circuits.

A third computer system consisting of a DEC 785 serves as a common meeting ground for the somewhat wider community of people concerned with integrated circuit design and manufacture. This computer provides the bulk of the text processing, printing, mail, simulation, and other facilities which are related to, but not directly concerned with, the actual operations in the integrated circuit processing facility. The 785 also provides a connection to an MIT Physical Plant computer and maintains an active alarm log for the building containing the integrated circuits fabrication facility.

Our initial CAFE (Computer Aided Fabrication Environment) software system was based upon the Berkeley Roving Shell with a file based data storage system. Our present CAFE system is based on the Berkeley system with an enhanced user interface menu and uses a commercial version of RTI INGRES data base system. Users log into this facility in the gowning room and then attach to the CAFE system at a terminal close by the equipment to be operated.

We have provided an ever growing number of functions to be accessed via the CAFE system. We have implemented a personal laboratory notebook facility. We have developed a standard forms based user interface mechanism which is used for most of our applications functions. Our implementation of an equipment reservation mechanism has been enthusiastically accepted by our laboratory users. We have implemented a number of transaction type applications such as defining of facilities and machines, starting lots, and reports which show machine status and the status of lots being fabricated. Just recently we have implemented an initial version of a process flow language and have provided two interpreters and a browser for this language. We are now trying to get people to use this process flow language on a routine basis.

During this period we installed two Sun 3/280 computers. One of these is used for the actual running of the CAF system for the fabrication laboratory, and the other is reserved for development of CAF system software. We have installed RTI INGRES on the Sun computers.

We have initiated the purchase of extra memory for the two Suns and also for the 785. We plan to augment each of these three CPUs to a total of 32 megabytes. Our 785 is often used by upwards of 30 users and extra memory is desirable in order to minimize page-outs. Both data base programs and lisp programs used for our process flow languages consume large amounts of memory and our plan is that the increased memory on the Suns will accommodate a sufficient number of lab users.

We have successfully ported all of the CAF software to the two Sun computers. While this entailed a significant amount of work, the programs are now much more portable than before. In addition we have ported all data residing in the previous multi-database system to RTI INGRES running on the Suns. We have almost completed the task of converting the last vestiges of our file based data storage scheme to RTI INGRES.

The performance increase, in speed, that resulted from this conversion to the Suns and RTI INGRES is approximately ten times. This is much appreciated by the lab users.

As all of our terminal concentrators use chaos protocols, it was necessary to install chaos network support in the Suns. We also had to make several increases in system resources in order to accommodate the larger number of lab users.

Previously we developed a generalized forms based user interface program, FABFORM. This single program, when called with a parameter file, produces a terminal display and allows a user to move from field to field and enter data. The type and content of the screen display is specified by an ASCII file which is referenced by data included in the parameter file. The form may have arbitrary length and the user can scroll up or down. At present, user interface commands are much like EMACS commands. When the user exits, or saves the data, an updated parameter file is written.

Using FABFORM, we have implemented a new version of the equipment reservations program which enables users to sign up for multiple machines at once. Users specify a list of machines that they will use in their next processing operation and access the data base for the current status of reservations. A locking mechanism has been devised so that multiple users cannot make conflicting entries. However different users may make reservations for different machines or different days. This program is in daily use and stores the sign up data in our data base.

We have also used FABFORM to implement a new multi-level menu system for the CAFE shell. We required a multi-level menu as we are accumulating quite a number of functions and were running out of space on the two level menu system that we ported from Berkeley. By using FABFORM we were able to implement this change fairly smoothly and also provide users with a more uniform interface.

We have continued to make substantial progress in the development of a data model and schema. Our system architecture is based on the Multibase system developed at CCA. This provides a uniform query interface to data residing in multiple autonomous, heterogeneous data bases.

Our data model is the functional model with support of extended data types including various temporal types as well as inexact, interval, and null values. The schema captures several important aspects of plant and

process management: fabrication facilities and equipment, users, equipment reservations, lots, lot tracking, wafers, process flow descriptions, WIP tracking, and lab activity information.

Primarily because of the interface layer between the application programs and the data base accesses we were able to change the data base out from under the application programs without requiring changes in those applications when we ported the data base to the Suns and RTI INGRES.

We have continued to progress on the development of a process flow language (PFL). The creation of a PFL and associated interpreters is the key to our approach for generating actual fabrication instructions and for collecting the data resulting from actual fabrication steps. The interpreters provide the actual meaning of the process flows expressed in the flow language. Our PFL development is based on a two-stage process step model which relates the goal of a change in wafer state first to the physical treatment parameters and finally to the actual machine settings used to process the wafers.

We have completed initial versions of walk through and fabrication interpreters and a browser all of which utilize our generalized forms based user interface program, FABFORM. The walk-through interpreter enables a process developer or potential user to see what will happen when the process is executed. The fabrication interpreter is similar, but in addition, allows for convenient entry of data as the wafers are actually processed. We have also completed an initial version of a past history browser which allows a user to review what actually happened in the fabrication of a lot of wafers. We have not yet been successful in getting these interpreters into daily use in the lab.

Spring 1988

This quarter, we made substantial progress in the development of our data model and schema. Our Gestalt system architecture provides a uniform query interface to data residing in multiple autonomous, heterogeneous data bases. Our functional data model provides support of extended data types including various temporal types as well as inexact, interval, and null values. We expanded our schema to represent more aspects of plant and process management: fabrication facilities and equipment, users, equipment reservations, lots, lot tracking, wafers, process flow descriptions, WIP tracking, and lab activity information.

The Gestalt data base interface routines were rewritten, expanded, and a new release (complete with an updated documentation paper) was made.

We have started a project to provide a schema display to let laboratory managers and users give more meaningful feedback about our schema.

We have completed a "data base walker" (DBW) program which enables application programmers (and others) to find their way around the existing data base. It displays an existing entity and allows the user to explore related entities. For example, one can display a facility and see that it has a list of machines. From there one can display a particular machine and see its attributes, etc.

We have written and released a new change status data entry program which is based on FABFORM.

We have developed, but not yet released, a generalized graphing program. This too, is based on FABFORM and allows the user to conveniently specify captions, axes labels, etc. This information, along with a data file is then provided to the locally developed "giraphe" programs which then produce the output graphs in a form suitable for terminal display or laser printer.

We have written and are now testing a generalized equipment uptime report generator. This is designed to provide the data to the graphing programs described above. It can produce graphs of uptime for a selected machine for a selected period of time or, alternatively, a summary of the relevant log entries relating to machine status changes.

A new program which interfaces with the Nanospec has been developed. It operates the Nanospec via the computer, initiates film-thickness measurements, and places the results in the appropriate field automatically, thereby reducing the operator interaction required.

We have initiated a project to interface the Gyrex mask maker directly to the computer. Presently users write data to magnetic tape and carry these tapes to the Gyrex.

We have initiated and made substantial progress on the development of a "hands off terminal." We chose a commercially available TI speech recognition module which plugs into an IBM PC/XT. Software has been developed to interface the TI PC software to control a FABFORM interface. As this speech recognition module is speaker dependent, the software automatically loads the data base appropriate to the login name. Several of us have "trained" the recognition software and the results are quite interesting. It remains to be seen if this recognition scheme is powerful enough to actually be useful in the fabrication laboratory, at least with this hardware.

We have continued to progress on the development of a process flow language (PFL). The creation of a PFL and associated interpreters is the key to our approach for generating actual fabrication instructions and for collecting the data resulting from actual fabrication steps. The interpreters provide the actual meaning of the process flows expressed in the flow language.

Our previous PFL development was based on only the machine setting view, in order to get something working as soon as possible. We now have a version of our PFL which is based on the two-stage process-step model which relates the goal of a change in wafer state first to the physical treatment parameters and finally to the actual machine settings used to process the wafers. We have recoded the CMOS baseline process in this new version and, in addition, have encoded a furnace monitor process which process is routinely used every week.

Besides a fabrication interpreter for this new version of the PFL, we have the rudiments of a simulation interpreter.

We have come to realize that we must provide for operation of partial flows. At least one impediment to the use of our PFL is that users change their minds about the process specification as they do the actual fabrication. By concatenating the processing history of fabrication with a number of partial flows we at least will have a trace which accurately reflects what happened.

We have made substantial progress on an expert PFL editor. This editor uses FABFORM as the user interface. Ideally one starts with an existing process flow, encoded in our lisp-like PFL syntax, which is somewhat similar to the desired process flow. The editor then displays this existing process flow with a forms based presentation and allows the user to modify the flow. The editor then produces the new flow encoded in PFL without the user even being aware of the lisp nature of the PFL. The editor supports the three views required by the two-stage generic process model and, in addition, allows any number of hierarchical levels of process flow definition.

Fall 1988

We made substantial progress in the development of programs relating to our data model and schema. Our Gestalt system architecture provides a uniform query interface to data residing in multiple autonomous, heterogeneous data bases. The Gestalt data base interface routines were expanded to include Lisp interface routines in addition to C interface routines by Ken Ishii.

Mike Ruf has completed his S.M. thesis proposal, "Management of IC Manufacturing Data," August 18, 1988. Mike worked on our CAFE project this past spring and is now at TI on his VI-A internship company assignment. His work is being done at TI. This thesis is an example of how our CAFE system data base philosophy is influencing related work at TI. A comprehensive set of data base tools, DBTOOLS, were generated by Ruf. These include:

- *dbinspect* - a "data base walker" program which enables application programmers (and others) to find their way around the existing data base. It displays an existing entity and allows the user to explore related entities. For example, one can display a facility and see that it has a list of machines. From there one can display a particular machine and see its attributes, etc.
- *dbcreate* - enable the creation of entities without writing a special application program.
- *dbquery* - implement data base queries.
- *dbmutate* - enable changes to be made to existing entities.
- *dbchoose* - select entities from the data base.

An S.B. thesis, "Schema Viewer: A Graphical Representation to Portray the Database Schema of the MIT CAFE System," was completed by Nazhin S. Zarghamee in May, 1988. This software provides a schema display so as to allow lab managers and users to provide more meaningful feedback as to the appropriateness and utility of our schema.

A number of application programs pertaining to status and log reports were written or modified by D. E. Troxel. These included *change-status* which is used for entering data relating to status changes, *describe-machine* to display current status, *new-machine* to enter a new machine into the data base, *format-equipment* to report the status of all equipment in a facility, *up* which is a generalized status and log report generator, and *dbt* which was used to transfer file based data to the data base. The *up* program can produce graphs of uptime for a selected machine for a selected period of time or, alternatively, a summary of the relevant log entries relating to machine status changes. The graphs can be output on a terminal or laser printer via *giraphe3* which was written by Duane Boning and Bob Harris.

A new program, *operate-machine*, was written by Mike McIlrath. This is initially used to make log entries when a machine is operated. It also provides a base for the creation of machine specific data entry and will be expanded substantially in the near future. In addition to being available from the CAFE menu, this program or procedure will be called by the fabrication interpreter in order to actually effect the machine operations. The *operate-machine* program is being expanded to create WIP entities when the program is exited without the finish time being specified. When this is accomplished several programs (*up*, *change-status*, and *describe-machine*) will have to be modified to include reports on active WIPs.

The next stage in the development of operate-machine is to make it specific to the actual machine being operated. This requires generation of opdesc entities which embody the description of the parameters or machine settings and the measurements or data which are to be collected.

A new program which interfaces with the Nanospec film thickness instrument has been developed. In addition to operating the Nanospec via the computer, it now initiates measurements and places the resulting thickness measurement in the appropriate field automatically, thereby reducing the operator interaction required to capture the measurement data for data base storage.

A program written by Peter Monta provides for an alternative direct interface between the Gyrex mask maker and a computer as opposed to requiring users to write data to magnetic tape and transport these tapes to the Gyrex.

We have developed and demonstrated a "hands off terminal." We chose a commercially available TI speech recognition card which plugs into an IBM PC/XT. Software has been developed by Peter Monta to interface the TI PC software to control a FABFORM interface. As this speech recognition module is speaker dependent, the software automatically loads the data base appropriate to the login name. Several of us have "trained" the recognition software and the results are quite interesting. This "hands off terminal" is now ready to be installed as the terminal next to the Nanospec in the fabrication laboratory.

We have continued the development of a process flow language (PFL). The creation of a PFL and associated interpreters is the key to our approach for generating actual fabrication instructions and for collecting the data resulting from actual fabrication steps. The interpreters provide the actual meaning of the process flows expressed in the flow language.

Our previous PFL development was based on only the machine setting view in order to get something working as soon as possible. We now have a version of our PFL which is based on the two-stage process step model which relates the goal of a change in wafer state first to the physical treatment parameters and finally to the actual machine settings used to process the wafers. We have recoded the CMOS baseline process in this new version and, in addition, have encoded a furnace monitor process which is routinely used every week.

We have made substantial progress on an expert PFL editor. An expert PFL editor has been completed by Rajeev Jayavant as his S.M. Thesis, "An Intelligent Process Flow Language Editor." This editor uses FABFORM as the user interface. Ideally one starts with an existing process flow, encoded in our lisp like syntax, which is somewhat similar to the desired process flow. The editor then displays this existing process flow with a forms based presentation and allows the user to modify the flow. The editor then produces the new flow encoded in our lisp like PFL without the user even being aware of the lisp nature of the PFL. The editor supports the three views required by the two-stage generic process model and, in addition, allows any number of hierarchical levels of process flow definition.

Our standard user interface, FABFORM, has been improved and extended so that it can now be called as a procedure from either C or Lisp.

We have made substantial progress on the development of a simulation interpreter. Duane Boning has completed his Ph.D. proposal, "Custom Fabrication Process Design: Tools and Methodologies," March 8, 1988. We have also realized that we must provide for operation of partial flows. One impediment to the use of our PFL is that users change their minds about the process specification as they progress with the actual fabrication. By concatenating the processing history of fabrication with a number of partial flows we at least will have a trace which accurately reflects what happened. A prototype version of the Suprem-III Simulation Interpreter has been completed. The interpreter generates Suprem-III fragments for multiple one-dimensional cross sections, produces a Makefile to minimize shared simulations, and provides analysis (plotting, sheet resistance, and threshold voltage) capabilities. Duane Boning is now working to get an accurate description of the CMOS Baseline process, so that meaningful comparisons of fabrication and simulation of the MIT standard defect array is possible, and so that realistic work with the flow language can progress.

We completed a substantial reorganization and cleanup of our CAFE software. We now have it all existing under a single directory, /USR/CAFE, in preparation for distribution to our fabrication laboratory and perhaps elsewhere.

We have acquired another computer and several workstations. We installed a VAX 750, garcon, for use as a file server. We have installed three monochrome and two color VS2000s, two Symbolics Lisp machines, and relocated the TI Explorer. We have installed a terminal concentrator in building 13. All of our computers are now on the same subnet with the result that file transfer and NFS services are now robust.

This summer we made a major new release of our CAFE software. We installed INGRES 5.1, an updated schema, *operate-machine*, and a series of applications programs related to status and log reports. These programs were described in previous progress reports. In addition, Mike Heytens generated a number of new database access routines which provide for more efficient filtering and sorting by implementing these operations at the underlying data base level instead of at the applications programming level. In spite of these optimizations we found that the system response time had deteriorated substantially both due to increased program size and an increase in the number of active users.

We temporarily took the computer CAF1.MIT.EDU used for CAFE development out of service in order to provide better response time on CAF2.MIT.EDU, which is the computer used by fabrication laboratory users. This doubled the memory available on CAF2 from 8 to 16 megabytes. Later, we were able to borrow some memory and we restored CAF1 to service and now have 16 megabytes of main memory for CAF2. We conducted timing tests on CAF2 with 16, 24, and 32 megabytes of memory. Several identical CAFE operations were started in quick succession. One sample operation implemented in Lisp took 160 seconds with 16 megabytes and 8 seconds with 24 or 32 megabytes. A different operation implemented in C took 125 seconds with 16 megabytes and 85 to 90 seconds with 24 or 32 megabytes. The primary reason that the Lisp program is slower than the C program is that it is much larger and the paging time is thus longer. Clearly we need to acquire more memory.

A fast method of logging in and out of the laboratory has been implemented by Rajeev Jayavant. The actual computations involved is not any faster but the user no longer must wait until they finish. This has been tested but not yet installed on CAF2.

An extension to the reservation program has been made and tested, though not yet installed on CAF2. A new program to make periodic reservations was written by Joseph Kaliszewski and he also modified the existing reservations program. Periodic reservation enables the convenient entry of lab hours and weekly scheduled preventive maintenance. Reservations indicated by the periodic reservation data structure are merged into blank areas of the reservation forms so as to provide this information to lab users. However, if a lab user wants to make a reservation which overrides these, he or she can do so.

We plan to re-implement the Gestalt object-like interface to provide an in-memory storage option for those sites and applications who do not want or need a persistent database, and do want and need fast, in-memory, object-like data structuring.

We have begun serious consideration of the integration of scheduling programs with CAFE. Xiewei Bai has started to integrate FABFORM into his factory simulation programs. We have discussed schema additions to represent the data required for scheduling programs but have not yet settled on their final form. The present approach is to write a single machine scheduler first with later extension to a multi-machine scheduler.

The programs relating to lot and wafer tracking have been rewritten and tested but not yet installed. *Create_lot* and *create_processname* were rewritten by Rajeev Jayavant. *Showlot* and *tracklot* were rewritten by D. E. Troxel. The latter two programs include optional reports of the history of opinsts and existing WIPs.

A project to automate the operation of the HP wafer prober has been initiated by Merit Hung and Joseph Kaliszewski under the direction of Professor Antoniadis. This project is proceeding in three phases, the first being development of a stand alone program on the computer supplied with the wafer prober. The second phase will be to enable the operation of this machine from another computer (CAF and/or CAF2). A third phase will be to integrate the operation of this machine with the *operate-machine* program running under CAFE.

III. Modular Process

The objective of this task was to develop and implement a workstation environment for process and device design. The motivation came from the success of the workstation environments for VLSI chip design, many of which were based on university work. Process and device design is much more compute-intensive than chip layout, because of the requirement for simulation of partial differential equations representing either the fabrication steps for wafers or the current flow in devices.

The progress of this activity is chronicled below by quoting from the semiannual reports prepared for each DARPA VLSI Contractor's Meeting, held each spring and fall.

Fall 1985

We have developed a preliminary architecture for a CAD program to implement modular process synthesis. The program, called MASTIF (MIT Analysis and Synthesis Tool for IC Fabrication), implements a workstation approach to IC fabrication process design. It is a menu- and window-oriented program which provides a methodology and uniform software structure for the connection of process- and device-design tools.

MASTIF is written in C and Fortran, which permits easy connection to existing and prospective process and device simulators and related tools. MASTIF currently supports incremental development and version management of a process specification under development. It also provides mechanisms for the simulation and analysis of physical cross sections, and includes interactive graphics interfaces to SUPREM-III and MINIMOS.

Spring 1986

The focus of this project is the development of methods for IC fabrication process and device design in the context of a Computer Aided Fabrication (CAF) research laboratory. The main thrust of our work centers on the development of what we call the MASTIF workstation. This is a menu- and window-oriented program written in C and FORTRAN which provides methodology and uniform software structure for the connection of process and device-design CAD tools. MASTIF currently includes a facility for incremental development and version management of a process description, management mechanisms for definition of physical cross sections deriving from the overall process description, and an interactive graphics interface and data interchange for process and device simulators (SUPREM-III and MINIMOS). With MASTIF the user can effectively develop and evaluate a fabrication process via a single integrated workstation.

In parallel with the development of MASTIF, work is under way in the area of process and device modeling. Extensive use of SUPREM-III has revealed several shortcomings of the program. Perhaps the most significant improvement that we have implemented in an experimental fashion is the inclusion of dynamic clustering for arsenic atoms in silicon. Similarly, we have scrutinized, found inaccurate, and improved significantly the electron-mobility model in the device simulator MINIMOS.

Two new process-simulation modules have been developed during this period. The first is used for the generation of as-implanted and diffused-doping profiles generated by means of Focused Ion Beam (FIB) implantation. This is in support of a separate FIB applications research program also supported by DARPA.

The second process-simulation module is still in an experimental form. It allows the simulation of nonuniform silicon oxidation. The calculated results are the shape of the grown SiO_2 and the stress exerted during the process at the Si/SiO_2 interface. A boundary-integral-equation method has been developed for the numerical solution of the problem. Because of the many uncertainties in physical parameters for grown SiO_2, the present simulator allows the investigation of three different oxide-movement behaviors: elastic, viscous, and visco-elastic. Work is under way to establish, by comparison with experimental data, the range of validity of these different models.

Fall 1986

Accomplishments for the period April through September 1986 are in three related areas, all involving CAD tools for process and device analysis. These include completion of the MASTIF project, modeling and simulation of kinetic clustering and precipitation effects during diffusion, two-dimensional silicon oxidation, and finally, investigation of a profile interchange format and database system.

The implementation of MASTIF was completed in May, 1986. The workstation incorporates tools for process specification and simple process verification, and interfaces (both front and back ends) to process and device simulation. Drawing on analogies from other areas of VLSI design, a blueprint for future development of tools beyond simple simulation has been completed [1]. During the summer, MASTIF was successfully ported to run on a true workstation, the Vaxstation-II.

On the pure process simulation side of the project, we have made an effort to adequately simulate the effects of impurity clustering and precipitation in silicon. The purpose was to implement models and methods previously developed [2] in a general and practical manner. A local version of the SUPREM-III [3] process simulator was modified extensively, and now incorporates a general exchange mechanism between mobile and immobile species of an impurity. Thus, both kinetic clustering and precipitation effects can be effectively simulated.

We have continued our effort on modeling two-dimensional thermal oxidation. A generalized Boundary Element Method (BEM) was developed for modeling linear viscoelastic materials. This new approach was based on Kelvin's solution modified for viscoelastic flow. It could handle a wide range of relaxation time, dealing with elastostatic deformation and viscous incompressible flow at the two extreme ends without anomalies as previously encountered. Simulations on Local Oxidation of Silicon (LOCOS) structures indicated that stress created during oxidation could be extremely high, particularly when the mechanical barrier effect of silicon nitride mask is included. This suggested that both silicon nitride and oxide flowed or plasticly deformed more readily than what we assumed. Stress-induced retardation of reaction rate and diffusivity of oxidants were also studied. In addition to an overall lowering of stress, the shape of the oxide changed significantly when such nonlinear effects were included.

One important finding of the MASTIF project has been to note the need for a uniform representation of wafer and device structure information, both for use by individual tools in a complete design system, and for interchange between different simulation sites. We have been investigating the form such an interchange format (PIF, or Profile Interchange Format) should take, as well as the need for standardization within the process and device simulation community. Preliminary proposals and implementations for both the "interchange" format and the corresponding "database" format have been completed [4].

SNC, a hierarchical storage system designed for process and device simulators, was operational. Two of the goals of SNC were to

(1) enable a program to generate a complex data structure for use by other programs. Hierarchical organization ensures that the structure can be expanded without introducing compatibility problems.

(2) aid program development by providing users a set of well-defined interface routines and run-time checking. With SNC, users need not create many output files. Most data can be put into a single SNC file. Retrieving data from a SNC file is simpler than from a normal ASCII or binary file.

A utility program was written for browsing SNC files. It could display the directory structure of a SNC file and print the contents. Access speed of SNC was made extremely fast by mapping the data file into program memory space.

References

[1] D. S. Boning, "MASTIF - A Workstation Approach to Integrated Circuit Process and Device Design," M.S. Thesis, MIT, May 1986.

[2] D. S. Boning, "Computer Simulation of Arsenic Clustering Effects in Integrated Circuit Processing," B.S. Thesis, MIT, May 1984.

[3] C. P. Ho and S. E. Hansen, "SUPREM III - A Program for Integrated Circuit Process Modelling and Simulation," Stanford Technical Report DEL 83-001, July 1983.

[4] D. S. Boning and T.-L. Tung, "A Proposal for A Profile Interchange Format," CAF Working Papers, MIT, April 1986.

Spring 1987

The implementation of the first version of MASTIF was completed in May 1986. A menu- and window-oriented program has been developed as the first step in meeting the need for an integrated process design system. MASTIF includes aids for process specification and simple process verification, and provides interfaces (both front and back ends) to the SUPREM-III process simulator and the MINIMOS device simulator. MASTIF was successfully ported to run on a true workstation, the Vaxstation-II under VMS. The program is now in its second phase of development and in the process of being ported to a UNIX environment.

We have continued our effort on modeling two-dimensional (localized) thermal oxidation. Thermal oxidation of silicon involves the diffusion of oxidant species from the gas-oxide interface to the oxide-silicon interface, and the transport of newly formed oxide away from the latter. Under suitable formulations, it can be shown that the diffusion process is a Laplace problem and the viscoelastic flow of oxide is a biharmonic problem. For these boundary value problems, the unknown boundary parameters can be obtained from the known boundary conditions without calculating the interior solutions. The diffusion problem is solved with a standard boundary element method (BEM) for potential problems. A generalized viscoelastic BEM has been developed to model the oxide flow. Utilizing constant-velocity kernel functions, this viscoelastic BEM can deal with a wide range of stress relaxation times, covering elastostatic deformation and incompressible creeping flow. Our approach achieves simplicity and efficiency by solving a two-dimensional problem as line integrals on the boundaries. Simulations of Local Oxidation of Silicon (LOCOS) structures indicated that stress created during oxidation could be extremely high, particularly when the mechanical barrier effect of silicon nitride mask is included. This suggests that both silicon nitride and oxide flow or plasticly deform more readily than assumed. Stress-induced retardation of reaction rate and diffusivity of oxidants are also studied. In addition to an overall lowering of stress, the shape of the oxide changes significantly when such nonlinear effects were included.

Development of general analysis tools and interfaces requires a central, agreed-upon representation of the structures these simulators and associated tools manipulate. We have been heavily involved in standards work related to a Profile Interchange Format (PIF), and are implementing a library of routines for accessing these structures based on the PIF. Specific tools and components under development include SNC, a local "database" form for PIF storage, PIFLIB, a library of routines for tool interface to the database, and PIFPLOT, a general, interactive structure analysis tool interacting with the PIF database. It is expected that the availability of a general representation for process and device structures will greatly enhance the capabilities of the MASTIF workstation, and will spark development of additional tools to aid in the design of IC processes.

Fall 1987

During the period of interest, substantial progress was made toward a computer-aided environment for process and device design. The two major aspects of this work were experimentation with a "Profile Interchange Format (or PIF) for the exchange of geometry and attribute information about simulated IC structures, and coupling between process design, simulation, and the development of a process flow language.

Earlier work with the MASTIF project has made it clear that work is needed to develop a common representation of simulated process and device structures in order to effectively interface and use existing TCAD tools. To this end, an international standards effort has been directed toward the specification of the Profile Interchange Format. A first-cut proposal for intersite exchange is on the table.

We felt it was important to experiment with this format before adoption as a standard. To this end, we modified the SUPREM-III and SIMPL-2 process simulators to output ASCII PIF code. An ASCII PIF parser was written, which reads the textual file into internal C structures. An interactive shell (the "PIF shell" or pshell) enables the user to browse the geometry and attributes, and to generate geometry or attribute plots.

We realized several benefits from this experimentation. First, we found several general and specific problems with the proposed PIF standard, and have made recommendations to the standard committee. Second, we found that the textual, ASCII "intersite" PIF format is inappropriate for direct use by various tools, and that an "intertool" or database version of the format is critical for effective use by CAD tools. We have formulated a plan for the coupled development and implementation of both an intersite and intertool version of the PIF.

The second aspect of this work relates to the interface between the rest of the CAF (Computer Aided Fabrication) environment and the TCAD environment. First, we have developed extensions of the PIF, which was originally developed in the context of process and device simulation only, so that we may represent cross sections and wafers in a uniform fashion for simulation, measurement, and manufacturing.

Secondly, we have been careful to consider the needs of process simulation during the development of the process flow language. We have built experimental interpreters of the flow language that provide input for the SIMPL-2 program (as a "change in wafer state" interpreter) and SUPREM-III (as a "wafer treatment" interpreter). We are working to enhance the interpreter, which in its current state might be best described as a "translator," so that we may effectively manage the design and simulation of the process flow.

Over the last six months, a few enhancements have been made to the two-dimensional thermal oxidation simulator. This simulator uses the boundary element method to model the viscoelastic flow of silicon dioxide and the diffusion of oxidants.

The program has been converted from VAX/VMS Pascal, which is not portable due to its language extensions, to C. It now runs on VMS and various Unix systems. Previous attempts of modeling nonlinear diffusion as a biharmonic system achieved only marginal success. We have switched to using domain cells, which are placed in critical regions only. Because only a few cells are used, the system matrix is smaller and computation is faster than the biharmonic approach. Better results have been obtained; also the anomaly of negative oxidant concentration has disappeared. It should be noted that the improvement is partly due to the use of "staggered grid" technique whereby different parameters are computed on different coordinate points.

The method for computing stress history has also been revamped. The "sources" are kept and updated at every time step to produce new stress fields. This arrangement permits stress relaxation time to be changed locally to simulate visco-plastic flow. The effectiveness of such a approximation still needs to be verified, pending numerical results for cylindrical oxidation mentioned below.

For his Ph.D. thesis at Stanford University, Dah-Bin Kao conducted a series of oxidation experiments on cylinders of different curvatures and proposed a viscous flow model for the effect of stress on oxide thicknesses. On convex structures, his model contains a positive feedback term that causes a self-consistent solution to blow up. The failure is partly due to the simplistic viscous flow model. A more realistic viscoelastic flow model will always guarantee the existence of a solution. We find that the new model still has an undesirable effect of freezing up, namely the oxide (fluid) suddenly solidifies when the curvature is below a critical value. We are seeking different functional behaviors of stress to match Kao's experimental data. This remodeling effort is necessary to provide parameters for use in the more general two-dimensional simulation.

Spring 1988

In the last six months, we have experimented with a "Profile Interchange Format" (or PIF) for the exchange of geometry and attribute information about IC structures. Prior work was based on an ASCII or "intersite" representation of the PIF, as recently outlined by Steve Duvall of Intel. Our previous experimentation with this ASCII PIF highlighted the unsuitability of the format for direct use by simulation tools.

We have recently been working, then, toward an "intertool" version of the PIF. An interface to Gestalt forms the basis of the "intertool" or "database" form of the PIF. Utilities based on the PIF database include a Suprem-III to PIF database (sup2db) for stuffing the results of process simulation into the database, and a database to Suprem-III (db2sup) for generation of the wafer structure as demanded by Suprem-III. In addition, we have developed a limited intersite PIF parser which translates the ASCII format into the database PIF (pif2db).

We are continuing development of the PIF database interface for direct use by a wide variety of process development tools, as well as pursuing implementation of tools based on this PIF database. The tools that must have access to wafer information, and thus to the PIF database, include not only process simulators such as Suprem-III, but also device simulators (such as MINIMOS), process flow language interpreters, grid manipulation programs, and analysis and plotting utilities.

We are continuing development of a CAD environment for the design of fabrication processes. This environment must be based on solid representations of the two "objects" being designed: structures to be fabricated, and fabrication processes. We have been working to lay the necessary representational groundwork upon which the complete CAD environment will be based: the PIF database provides the framework for the representation of the wafer, while the Process Flow Language (PFL) provides the representation of the process. In the last six months, these representations have been developed to the point where we were able to begin experimentation with tools based on the PIF and PFL.

We have developed an experimental Process Simulation Manager, and coupled this with the need to simulate the MIT Baseline CMOS process. The baseline process was represented in our experimental Process Flow Language. A nominal "Suprem-III Translator" produces fragments of Suprem-III code to simulate operations within the flow. Using UNIX utilities, particularly make, the prototype Manager enables us to maximize the sharing of computation between multiple cross sectional simulations, as well as minimize and automate resimulation when the process changes. A number of post-processing utilities allow the interactive analysis of final and intermediate simulated profiles.

The prototype Simulation Manager does not provide a tight coupling between the flow and simulation, nor does it manage other important aspects of design besides simulation. We are continuing development of the Simulation Manager, as well as a Design Supervisor to provide additional capabilities in process verification, analysis, and synthesis.

Fall 1988

Development of the Profile Interchange Format (PIF) has proceeded in two areas: implementation of a PIF program interface, and development of individual PIF utilities. Taking the published proposal for the "intersite" (or ASCII) PIF as a starting point, we have been developing a set of routines through which CAD and CIM programs may access PIF objects. This "intertool" format is based on the Gestalt database interface to a geometric and attribute schema definition tailored to the PIF, and provides a uniform or "standard" functional program interface. Tools implemented with this interface should, like the data itself, thus be portable. We have so far developed a small set of such utilities, including format conversion programs from SUPREM-III to the PIF ("sup2pif" and "pif2sup"), between the ASCII and database versions ("pif2db" and "pifdump"), as well as a simple profile plotting utility ("pifplot"). Implementation of the interface and specific PIF utilities is continuing.

The Technology (encompassing both Process and Device) CAD Environment is based solidly on the wafer representation (PIF) and the process representation (PFL or Process Flow Language). During the last six months, we have focussed on development of two tools for this environment. The first of these is a prototype Simulation Manager, which has as a key component a translator from the PFL representation of the process to the input required by SUPREM-III. The specification and handling of multiple one-dimensional cross sections for simulation has been a major addition to the Manager. Utilities allow minimal simulation of the process (the MIT CMOS Baseline process has been the vehicle for testing the manager), evaluation of where the process diverges for multiple cross sections, as well as interactive examination of simulation results.

The second area of research we have begun is to investigate simple process synthesis utilities. We have written initial versions of an "Oxidation Advisor," an "Implantation Advisor," and a "Diffusion Advisor" to provide physically-based initial guesses for process parameters during process design. Work is underway to further build "Correction Advisors," as well as to experiment with physically-based optimization methods.

IV. Scheduling

The objective of this task was to develop algorithms to be used in scheduling operations in a VLSI fabrication facility. Suitable models were developed, that captured the essential features of the steps for scheduling purposes. Interesting challenges included the importance of setup times, frequency of equipment failures, and use of the same machine more than once in a single process flow.

The progress of this activity is chronicled below by quoting from the semiannual reports prepared for each DARPA VLSI Contractor's Meeting, held each spring and fall.

Fall 1985

During this period, we have begun to gather information on the scheduling issues in semiconductor fabrication facilities. We plan to develop a model of such systems which will allow the development of scheduling algorithms that incorporate feedback of state information. Such models will include phenomena such as machine failures, demand variations, set-up times, parts mix, maintenance, finite machine capacities, and other features. These very features are the ones that frequently make system scheduling difficult.

It is anticipated that the management system will be organized hierarchically. Short-term management and control -- real-time control -- must be consistent with long-term management decisions. Lower levels in the hierarchy are more spatially (as well as temporally) limited than higher levels. That is, they deal with fewer machines. Higher levels have more power than lower, in the sense that they set objectives for the lower levels, which must indicate their capabilities to the higher levels.

Spring 1986

Research during this period focused on three activities: studying the integrated-circuit fabrication process at a systems level, formulating a mathematical model of an integrated-circuit fabrication facility, and developing an electronic sign-up sheet for scheduling the laboratory.

The first study is to define the scheduling problem. This involves studying VLSI technology, the construction of the MIT Integrated Circuits Laboratory, and user requirements. It was discovered that a few key elements of the problem may have dramatic impact on the final schedule. A setup is required when there is an important change in type of operation at a machine, and can be due to changes of gases, water temperatures, or other required materials. Setups complicate scheduling because machines that are undergoing such changeovers are not available for productive operations. There is an incentive to put off changeovers as long as possible because when they take place often, production capacity is reduced. On the other hand, if changeovers are performed too infrequently, some lots will be delayed a long time.

Randomness is due to machine failures and unpredictable demand on the facility. In a university setting, the mix of chips going through the facility has very little consistency from week to week. We will have to group the demands for scheduling purposes, and use guesses based on historical data. The presence of randomness means that any schedule must have some idle time in it, so that a perturbation does not propagate indefinitely. It also means that there must be some means for rescheduling after events like machine failures.

We are also investigating the differences between a university laboratory and a commercial production system. In a commercial facility, machine failures are no less important, but there is a more predictable demand on the facility. The CAF Scheduling group visited DEC in Hudson, Massachusetts on February 11, 1986 to observe their production system.

The second activity is to develop a scheduling model of the system. Developing such a model requires identifying scheduling objectives, decomposing the problem, solving each of the subproblems, developing a database to store the required information, and testing the solution on the facility. Scheduling is complicated by setups and randomness, and various sources of both are being studied. The main focus of the analytic activity is

the setup issue. The approach now being explored views setups as events that take place at a lower frequency than machine operations and most other events. A slow-time-scale model is being explored that accounts for the time that the system spends in each setup configuration, as well as the time spent in changing configurations. The impact of setups on capacity is modeled explicitly.

Remaining work on this formulation includes modeling the scheduling objective, and then translating the setup frequencies that are calculated this way into actual times at which to perform the changes in configurations.

The purpose of this activity is to develop a simple Version 1 scheduler which will not use the sophisticated ideas that are described above, but which can be implemented and used relatively quickly. The existing paper system is being designated as Version 0, and the first electronic system as Version 1. Great consideration is being given to ease of use. It will be designed to emulate the paper sign-up sheet in the old facility, with no additional features except for the fact that it can be accessed from any terminal.

Fall 1986

Research during this period focused on three activities: Studying the integrated circuits fabrication process at a systems level; Formulating a mathematical model of an integrated circuits fabrication facility; and Developing simulation and scheduling software.

The effort to define the scheduling problem continues. We continue to study MIT laboratories and commercial facilities. We are investigating the following phenomena: setups, machine failures, random demands, and other random, discrete events that complicate scheduling.

A model formulation is being developed that takes into account all the kinds of events described above. A combinatorial optimization approach would be infeasible because of the problem size, but, because the important events take place at different frequencies, there is structure that can be exploited. A multiple time scale decomposition is being explored.

Particular emphasis is being devoted to the long time that wafers spend in clean rooms. The response to random events is influenced by the fact that a long time passes between the introduction of material into the system and when it encounters later operations.

We have begun an effort to test the ideas and model formulation described above. Based on our study of MIT and other facilities, and on the model summarized above, we are planning a multi-level scheduling program. The top levels will be based on a continuous formulation that emphasizes delay, and the lower levels will perform simplified searches.

A demonstration software designed to schedule the Integrated Circuits Laboratory at MIT is being developed. It makes use of an Artificial Intelligence approach to facilitate the search. The possibility of resolving incomplete, inconsistent and qualitative information as well as learning through experience are also being investigated.

Spring 1987

In the effort to define the scheduling problem, we are concentrating on using the MIT laboratories as case studies. Mathematical and simulation models, described below, are being based on what we learn here.

As a mathematical model of an IC fabrication facility, the multiple time scale decomposition under development shows great promise. A new basic model is being investigated which will help us refine and better justify the tentative mathematical results we have developed thus far on hierarchical scheduling.

In this approach, the scheduling algorithm is divided into a set of levels which correspond to classes of events that are distinguished by their frequencies. At each level, two kinds of calculations are performed: small

linear programs, to determine frequencies of higher frequency (lower level) events; and simple combinatorial optimizations, to determine exact times for the events of that level, whose frequencies have been calculated at higher levels.

Software which will implement a multiple time scale decomposition approach to hierarchical scheduling is under development. A simulation is also under development. The scheduling software will first be tested with the simulation, and then used to run the laboratory.

An electronic machine reservation system is also under development. In its initial version, it is essentially an electronic sign-up sheet for equipment. It will later be used, after modifications, as the lowest level of the hierarchical scheduler.

Fall 1987

During this period, the theoretical basis for the scheduling and planning hierarchy was strengthened.

Three kinds of aggregation/decomposition are required for a fully developed scheduling and planning hierarchy: temporal (in which higher level decision-makers make less frequent but individually more important decisions); spatial (in which higher level decision-makers affect a larger portion of the facility); and material (in which higher level decision-makers are concerned with broader classes of products).

Temporal decomposition divides the scheduling and planning problem along time scale or frequency lines; spatial decomposition puts boundaries in space around groups of pieces of equipment. Our goal is to fully develop all three parts of the hierarchy, and to implement them in a set of computer programs which will be integrated with the CAFE system. These programs, and their communications, are likely to be complex, but they offer great savings in computer time and accuracy in comparison to other methods.

Most progress has been made in the temporal hierarchy. A structure has been developed based on a set of necessary conditions described in VLSI Memo 87-406. These conditions establish the data of interest at each level of the hierarchy, as well as the relationships among the data at different levels of the hierarchy. These relationships in turn suggest the form of the computations that are required at each level.

Specific classes of events that we have been studying include operations, machine failures setup changes, and demand variations. In present formulations, we assume that each class of events occurs at a frequency which is substantially different from all the others, and each class is treated in detail at a different level of the hierarchy.

We have also studied spatial decomposition by performing numerical experiments. We conclude that, at each level of the hierarchy, local decisions should be made on the basis of the amount of material in the system, the amount of material found locally, and the difference between actual behavior and the requirements specified at higher levels. Some ideas for combining the spatial and temporal decompositions are beginning to emerge.

We have identified the roles of different kinds of users of the hierarchy (students or technicians; professors or engineers; managers; and scheduling system implementors). They will interact with the programs at different levels in different ways.

We have been designing a set of simulation programs that will have a variety of uses, including the testing of the planning and scheduling methods developed here and the prediction of the actual behavior of a production system for managerial purposes. These programs will carefully separate the scheduling/planning programs from the model of the plant. This will allow easy comparison of different schedulers, and will facilitate transferring the scheduler to a real fabrication facility.

Spring 1988

A draft report has been written summarizing our knowledge of semiconductor fabrication from the point of view of a scheduler. We have sent it to many well-informed people in industry and academia and solicited their comments by including a questionnaire. Some comments have come back, but we expect to get the bulk of the responses during April 1988. Readers who would like to review this draft and possibly make comments are urged to contact us.

The multiple-time-scale decomposition is under development and shows great promise. A new basic model is being investigated which will help us to refine and better justify the tentative mathematical results we have developed thus far on hierarchical scheduling

In this approach, the scheduling algorithm is divided into a set of levels which correspond to classes of events that are distinguished by their frequencies. At each level, two kinds of calculations are performed: small linear programs, to determine frequencies of higher frequency (lower level) events; and simple combinatorial optimizations, to determine exact times for the events of that level, whose frequencies have been calculated at higher levels.

We are devoting a great deal of attention to the scheduling of setups since they are likely to become important in modern multiple-purpose fabs, i.e., those that can be used for more than one basic process.

Software which will implement this multiple time scale decomposition approach to hierarchical scheduling is under development. A simulation is also under development. The scheduling software will first be tested with the simulation, and then used to run the laboratory.

Fall 1988

An important phenomenon was added to the set of models that have been part of our study, namely the fact that many semiconductor fabrication operations are performed on a batch of wafers simultaneously. Examples include oxidation, deposition, ion implantation, and others. While this is seen in other kinds of manufacturing, it is pervasive in this industry.

This feature is important because in order to realize the full capacity of a system, the machine chambers must be as full as possible. There are two reasons for this: (1) it takes as much time to do an operation on one wafer as 100, but 99% of the capacity is wasted if only one wafer is in a chamber that can hold 100, and (2) maintenance must be performed on machines that do deposition operations when the total amount deposited since the last maintenance reaches a given level, independent of the number of wafers that were in the chamber when the depositions took place.

One way to keep the chambers full is to have large lots. However, this is not desirable in a system that has low volume or that has a diversity of products. Another approach is to group together distinct wafers that require the same operations. This leads to more complex modeling and scheduling issues, which we are currently studying and simulating. A prototype scheduler has been built for a single machine.

V. Equipment Models

This task began in Fall 1986. Its objective was to develop models for semiconductor fabrication equipment that were capable of being used for several purposes, and to integrate those models into CAFE. In fact, the integration into CAFE was delayed until the present contract expired. The principal results were the development of models incorporating a mix of empirical data and fundamental reasoning, and a demonstration that such models could be used to optimize the operating point of practical equipment. The first process step considered was LPCVD (Low Pressure Chemical Vapor Deposition) of undoped polysilicon.

The progress of this activity is chronicled below by quoting from the semiannual reports prepared for each DARPA VLSI Contractor's Meeting, held each spring and fall.

Spring 1987

VLSI machine modeling has taken a concrete form during the past several months. The thrust of the work is to combine analytical modeling with matrix experimental approaches and to provide an executable program which will facilitate the following:

1. Off-line quality control to determine the point in operating space which provides the greatest robustness against variations in process parameters and therefore yields the most consistent results.

2. On-line quality control used to tune a process based either on measurements made after a previous run or on in-situ measurements.

Matrix experimentation is a collection of methods wherein some or all of the relevant process variables are varied simultaneously in an experiment and information is extracted from the results statistically [1,2]. In contrast to conventional single-variable experimentation, the matrix approach offers a tremendous economy of experimental effort. This is crucial in a production environment as the interruption to work flow must be kept to a minimum, while in a research environment it is useful in order to optimize a new process as quickly as possible. Matrix experimental techniques have been employed with great success on VLSI processes at AT&T Bell Laboratories for approximately the last six years [3,4].

Our process modeling effort will begin by utilizing analytical and experience based background to define the process variables and their values for a matrix experimental approach. The analytical work will be based both on process models available in the literature and on simple physical models. Future extensions of the work will attempt to more closely couple the analytical and experimental approaches by using matrix experimentation to verify analytical models, specify numerical values for the analytical models, and also to improve analytical models.

The first process selected for the machine modeling effort is LPCVD of polysilicon. Optimization of thickness uniformity across a wafer, between wafers in a single lot, and between runs will be the goal.

References

[1] G. Box, W. Hunter, and J. Hunter, "Statistics for Experimenters and Introduction to Design Data Analysis and Model Building," Wiley, 1978.

[2] G. Taguchi, "Introduction to Quality Engineering," Krauss International Publications, White Plains, NY, 1986.

[3] M. Phadke, N. Kuckar, D. Speeney, and M. Grieco, "Off-Line Quality Control in Integrated Circuit Fabrication Using Experimental Design," The Bell System Technical Journal, 1983.

[4] R. Nackar and A. Shoemaker, "Robust Design: A Cost-Effective Method for Improving Manufacturing Processes," AT&T Technical Journal, March/April, 1986.

Fall 1987

During this period significant progress was made by George Prueger, a master's candidate, on the modeling of LPCVD of undoped polysilicon. The principal area of progress was on the experimental side of the project. An orthogonal-array experimental design was created using two flow rates, an injector position, and pressure as the process parameters. Arrangements were made with BTU Bruce of Billerica, Massachusetts to conduct experiments on equipment in their laboratory. A complete experimental regimen was established with BTU's full cooperation.

The experimental work consisted of running lots of 150 six-inch wafers, of which thirteen were test wafers and the remainder reused dummy wafers. Approximately 5000 Å of poly was deposited in each experimental run. Thickness measurements were made at five spots on each wafer using optical measurement techniques. Data was plotted for each of the nine experiments of the orthogonal array.

Several of the data points were repeated in an effort to examine consistency of results. A second smaller array of experiments was conducted using only the load-end injector (the other two injectors were turned off) in a depletion type experiment. These experiments were designed to yield information on the rate constants of the chemical reactions.

Analytical modeling using a finite difference model incorporating both diffusion and convection in the system has begun.

Spring 1988

During this period the first equipment model was completed. This model concerns the low pressure chemical vapor deposition of polysilicon in a horizontal tube furnace. The model consists of a one dimensional finite difference numerical formulation which encompasses convective and diffusive mass transport in the annular space between the wafers and the tube liner, and accounts for the surface reaction rate limited deposition of polysilicon on the wafers with the associated generation of hydrogen and incorporation into the bulk of the gas. The model permits as input the gas flowrate to the three injectors, positions of the injectors, reactor geometry, temperature profile down the tube, and operating pressure. The adjustable coefficients in the model have been calibrated using a series of designed experiments performed at the applications lab of BTU Bruce of Billerica, Massachusetts. In these experiments, 150 wafer loads of six-inch wafers were used. The experiments involved four parameters, two gas flowrates, one injector position, and operating pressure.

The results from the model are in excellent agreement with the experimental work. The model appears to predict the profile of growth rate down the tube accurately. The model is accurate enough to be interrogated for process optimization, and gives a predicted optimum set of process parameters which is very close to that found by the experimental Taguchi optimization.

In the near future, the model will be tested in on-line quality control for its ability to predict changes around an operating point.

During this period, George Prueger has completed his master's thesis on LPCVD of poly. A paper is in preparation for submission to the Journal of Semiconductor Manufacturing.

Also during this period, Michele Storm has investigated the use of a software package called ULTRAMAX for on-line quality control in CAF.

Fall 1988

During this time period, progress was made on three projects, and a fourth project was initiated. The first ongoing project is an equipment model for the LPCVD of doped polysilicon. The second ongoing project concerns the use of dimensional analysis in the design of experiments. The third ongoing effort concerns the

use of sequential design of experiments for process optimization and control. The new project concerns the development of a general framework for process control.

A common element in our equipment modeling efforts is the fusion of statistical design of experiments with physically based mechanistic modeling. Our first completed work is an equipment model for LPCVD of polysilicon in a horizontal tube furnace. In this work, George Prueger developed a finite difference model for the equipment and process and calibrated that model using data derived from designed experiments. Our current projects on modeling of doped poly and the use of sequential design of experiments seek to further the effort at the combination of experimental design and physical knowledge.

The project on modeling of the LPCVD doped poly is being carried out by Master's student Parmeet Chaddha with experimental support coming from the BTU applications lab, in Billerica, MA. The goal of this work is to develop an equipment model which aids in the design of the cage in the doped poly process. The function of the cage is to serve as a deposition site for reactants formed in the annular layer between the cage and the tube walls, thereby resulting in uniform thickness deposition on the wafers. The challenge in cage design is to specify the size and distribution of the holes so that the deposit is uniform on the wafers, and the deposition rate is as high as possible. Our approach is to develop a highly simplified analytical model which predicts deposition uniformity and rate as a function of cage geometry, and to calibrate this model using data derived from designed experiments. Roughly one half of the experiments have been completed during this reporting period, with the work being performed on 4 inch wafers at BTU, Inc.

Master's student William Wehrle has been developing methods of using dimensional analysis in the design of experiments. Dimensional analysis is a relatively easy means of deriving relationships between variables based on physical arguments. A very general theorem called the Pi Theorem provides a set of rules which can be followed to create the dimensionless groupings which characterize a problem. The theorem, however, results in a tremendous choice of sets of these groupings. Mr. Wehrle has developed two rules which narrow the choice down when the Pi theorem is applied to the design of experiments. He will test the theory in application to the LPCVD of LTO.

Master's student Michele Storm is working on the use of sequential design of experiments for process optimization and control, in collaboration with Ultramax Corp. of Cincinnati, OH. Sequential optimization is a technique particularly well suited to manufacturing, where processes are inherently sequential. The basis of the method is to perform a local regression to the measured data, thereby calibrating a quadratic response surface centered near the current point of operation. This response surface may be considered to be a Taylor series representation of the process. The polynomial representation is then used to design the next experiment in a continuous cycle of "learning and advising." Our goal is to use dimensional analysis to create grouped variables for use in the models, as a replacement for the primitive variable currently used. Ms. Storm has written a sequential optimization program and will shortly begin experimental work on wire bonding. She will compare the effectiveness of grouped (dimensionless) variables with the effectiveness of primitive variables for optimization of wire bonding.

Doctoral student Ruey-Shan Guo was hired toward the end of the summer (August 1988). Mr. Guo has spent the intervening months acquiring a background in statistics and experimental design.

VI. Mechanical Property TCAD

This task was initiated in the last year of the contract. The objective was the development of TCAD tools which could predict the mechanical, as opposed to electrical, properties of integrated circuits and structures. Particular attention was paid to residual stress, because of its importance in some catastrophic failure mechanisms. The task was undertaken because of its importance for device reliability, and was relevant to the objectives of this contract because of the overlap of many of the equipment and process models in CAFE, and the attractiveness of incorporating the concepts into CAFE and into the process-design workstation.

The progress of this activity is chronicled below by quoting from the semiannual reports prepared for each DARPA VLSI Contractor's Meeting, held each spring and fall.

Spring 1988

This project is directed toward the development of Technology CAD (TCAD) tools with which to predict the mechanical behavior of microfabricated devices. The application of these tools will be twofold: prediction and modeling of microsensor and microactuator devices as part of the design process; and analysis of stress distributions in microelectronic parts for reliability assessment. There are many well-documented examples of device failure produced by mechanical failure (cracking of dielectrics and conductors, and delamination of coatings). At present, these are handled on a case-by-case basis, and only when the fault is detected during life test of finished parts. The goal here is to use research on test structures to build a data base and CAD environments with which to model the stress distributions in microelectronic structures prior to fabrication in order to identify high-risk sections of a design that might be prone to catastrophic mechanical failure. In addition, these tools, developed initially with the use of experiments on deformable structures, such as beams, cantilevers and diaphragms, will be used for predictive modeling during the design of sensors and actuators that include such deformable components. Effort to date has concentrated on developing data on the stress and modulus of polysilicon as a function of its processing history.

Fall 1988

The overall goal of this project is the development of a design capability that includes the mechanical properties of microelectronic materials. The specific goal during this past year has been to develop experimental methodologies with which to determine the residual stress in microelectronic thin films. The material selected for the first set of experiments was polycrystalline silicon, because it is well known that its residual stress is very dependent on process conditions. Several experiments were carried out on polysilicon as a function of its deposition and doping. Using techniques of surface micromachining, suspended polysilicon beams and cantilevers were fabricated in various thickness, doped variously, and subjected to different sequences of thermal annealing. The residual stress in the polysilicon and the stress uniformity were then determined by examining vertical deflection of cantilevers with optical interferometry. In addition, the maximum free-standing non-buckled beam length was determined, which can also be related to residual stress. The detailed descriptions and conclusions are contained in two papers which have resulted from this work, which are attached to this report.

VII. Publications

Here is a list of publications based on work supported in part by this contract. Those readers interested in more detail on any particular topic may consult these publications. Single copies of MIT VLSI Memos for personal use may be obtained without charge (supplies permitting) by writing to the Microsystems Technology Laboratories, Room 39-321, MIT, Cambridge, MA 02139.

PUBLICATIONS

Duane S. Boning and Dimitri A. Antoniadis, "MASTIF - A Workstation Approach to Fabrication Process Design," *Digest of Technical Papers, IEEE International Conference on CAD, ICCAD-85*, Santa Clara, California, November 18-21, 1985, pp. 280-282; also MIT VLSI Memo No. 85-259, September 1985.

T.-L. Tung and D. A. Antoniadis, "A Boundary Integrated Equation Approach to Oxidation Modeling," *IEEE Transactions on Electron Devices*, vol. ED-32, no. 10, pp. 1954-1959, October 1985.

Christopher Lozinski and Stanley B. Gershwin, "Dynamic Production Scheduling in Computer-Aided Fabrication of Integrated Circuits," *Proceedings, 1986 IEEE International Conference on Robotics and Automation*, San Francisco, California, April 7-10, 1986; also MIT VLSI Memo No. 86-304, March 1986.

Stanley B. Gershwin, "Stochastic Scheduling and Set-Ups in Flexible Manufacturing Systems," *ORSA/TIMS/Conference on Flexible Manufacturing Systems: Operations Research Models and Applications*, Ann Arbor, Michigan; also MIT VLSI Memo No. 86-314, May 1986.

Duane S. Boning, *MASTIF - A Workstation Approach to Integrated Circuit Process and Device Design*, M.S. Thesis, MIT, Department of Electrical Engineering and Computer Science, May 1986; also MIT VLSI Memo No. 86-325, June 1986.

Sheldon X. C. Lou, Garrett Van Ryzin, and Stanley B. Gershwin, "Scheduling Job Shops with Delays," *Proceedings, 1987 IEEE International Conference on Robotics and Automation*, Raleigh, North Carolina, March 31-April 3, 1987, vol. 3, pp. 1345-1349; also MIT VLSI Memo No. 87-368, March 1987.

Thye-Lai Tung, Jerome Connor, and Dimitri Antoniadis, "A Viscoelastic BEM for Modeling Oxidation," *Fundamental Research on the Numerical Modelling of Semiconductor Devices and Process, Proceedings from NUMOS I, The First International Workshop on Numerical Modelling of Semiconductors*, Los Angeles, California, December 11-12, 1986, vol. 9, pp. 115-121; also MIT VLSI Memo No. 87-369, March 1987.

Oded Z. Maimon and Stanley B. Gershwin, "Dynamic Scheduling and Routing for Flexible Manufacturing Systems that have Unreliable Machines," *Proceedings, 1987 IEEE International Conference on Robotics and Automation*, Raleigh, North Carolina, March 31-April 3, 1987, vol. 1, pp. 281-288; also MIT VLSI Memo No. 87-370, March 1987.

Oded Z. Maimon and Stanley B. Gershwin, "Dynamic Scheduling and Routing for Flexible Manufacturing Systems that have Unreliable Machines," *Operations Research*, vol.. 36, no. 2, pp. 279-293, March-April 1988; also MIT VLSI Memo No. 87-370, March 1987.

Stanley B. Gershwin, "A Hierarchical Framework for Discrete Event Scheduling in Manufacturing Systems," *Springer-Verlag Lecture Notes in Control and Information Sciences*, vol..103, pp. 197-216, August 1987; also MIT VLSI Memo No. 87-406, September 1987.

Stanley B. Gershwin, "A Hierarchical Framework for Discrete Event Scheduling in Manufacturing Systems," *IIASA Workshop on Discrete Event Systems: Models and Applications*, August 3-7, 1987, Sopron, Hungary; also MIT VLSI Memo No. 87-406, September 1987.

Stanley B. Gershwin, "A Hierarchical Framework for Manufacturing Systems Scheduling: A Two Machine Example," *Proceedings, 26th IEEE Conference on Decision and Control*, Los Angeles, California, December 9-11, 1987,vol 1, pp. 651-656; also MIT VLSI Memo No. 87-409, September 1987.

E. Sachs and G. Prueger, "Process Model Testing and Optimization Using Matrix Experimentation," *Proceedings, ASME Winter Annual Meeting*, December 1987; also MIT VLSI Memo No. 87-405, September 1987

E. M. Sachs, G. Prueger, and R. Guerrieri, "Process Model Testing and Optimization Using Orthogonal Array Experimentation," presented at the *1988 Winter Annual Meeting American Society of Mechanical Engineers*, Boston, Massachusetts, January, 1988, PED vol. 27.

T.-L. Tung, J. Connor and D. A. Antoniadis, "A Boundary Element Method for Modeling Viscoelastic Flow in Thermal Oxidation," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 7, no. 2, pp. 215-224, February 1988; also MIT VLSI Memo No. 87-397, July 1987.

Emanuel Sachs and George Prueger, "Process Model Construction and Optimization Using Statistical Experimental Design," *Proceedings, Manufacturing International 88*, Atlanta, Georgia, April 17-21, 1988; also MIT VLSI Memo No. 88-442, March 1988.

D. S. Boning and D. A. Antoniadis, "A Workstation Approach to IC Process and Device Design," *IEEE Design and Test of Computers*, vol. 5, no. 2, pp. 36-47, April 1988.

N. S. Zarghamee, *Schema Viewer: A Graphical Representation to Portray the Database Schema of the MIT CAFE System*, S. B. Thesis, MIT, Department of Electrical Engineering and Computer Science, May 1988.

T. A. Lober, J. Huang, M. A. Schmidt, and S. D. Senturia, "Characterization of the Mechanisms Producing Bending Moments in Polysilicon Micro-Cantilever Beams by Interferometric Deflection Measurements," *Proceedings, IEEE Solid-State Sensors Conference Workshop: Technical Digest*, Hilton Head Island, South Carolina, June 6, 1988; also MIT VLSI Memo No. 88-490, November 1988.

R. Jayavant, *An Intelligent Process Flow Language Editor*, M. S. Thesis, MIT, Department of Electrical Engineering and Computer Science, September 1988; also MIT VLSI Memo No. 88-475, September 1988.

J. Huang, T. A. Lober, M. A. Schmidt, and S. D. Senturia, "The Maximum Free-Standing Length of Polycrystalline Silicon Microbeams," *Proceedings, International Conference on Material and Process Characterization*, Shanghai, China, October 1988.

Stanley B. Gershwin, "Hierarchical Flow Control: A Framework for Scheduling and Planning Discrete Events in Manufacturing Systems," *Proceedings, IEEE Special Issue on Discrete Event Systems*, Revised VLSI MIT Memo 87-406; also MIT VLSI Memo No. 88-482, October 1988.

M. L. Heytens and R. S. Nikhil, "GESTALT: An Expressive Database Programming System," submitted to the *SIGMOD Record*; vol. 18, no. 1, March 1989, pp. 54-67; also MIT VLSI Memo No. 88-484, November 1988.

T.-L. Tung, *Boundary Element Techniques for Modeling Thermal Oxidation of Silicon*, Ph.D. Thesis, MIT, Department of Electrical Engineering and Computer Science, September 1988; also VLSI Memo No. 88-489, November 1988.

G. H. Prueger, *Equipment Model for the Low Pressure Chemical Vapor Deposition of Polysilicon*, M.S. Thesis, MIT, Department of Mechanical Engineering, MIT, March 1988; also, MIT VLSI Memo No. 88-485, November 1988.

E. Sachs, G. Prueger, and R. Guerrieri, "Equipment Models for Process Optimization and Control Using Smart Response Surfaces," *Proceedings, Electrochemical Society*, Chicago Illinois, October 1988; also MIT VLSI Memo No. 89-513, March 1989.

INTERNAL MEMORANDA

Stanley B. Gershwin, "Dynamic Production Scheduling in Computer-Aided Fabrication," MIT VLSI Memo No. 85-243, April 1985.

Thye-Lai Tung, Jerome Connor, and Dimitri A. Antoniadis, "A Boundary Element Method for Modeling Viscoelastic Flow in Thermal Oxidation," MIT VLSI Memo No. 87-397, July 1987.

Emanuel Sachs and George Prueger, "Process Model Testing and Optimization Using Matrix Experimentation," MIT VLSI Memo No. 87-405, September 1987.

Robert M. Harris and Duane S. Boning, "GIRAPHE V3.3: A User's Manual with Examples," MIT VLSI Memo No. 88-487, November 1988.

Rajeev Jayavant, "A User/Programmer Guide to the Fabform User Interface," MIT VLSI Memo No. 88-487, November 1988.

X. Bai and S. B. Gershwin, "A Manufacturing Scheduler's Perspective on Semiconductor Fabrication," MIT VLSI Memo No. 89-518, April 1989.